

# GPGPU with OpenCL, CUDA and DirectX - a comparison

Adam Söderström 19930327-3750  
Adaso578@student.liu.se

---

## Background

During the last couple of years the inclination of the performance of central processing units (CPU) has decreased. The reason for this is because the theoretical limit of the transistors clock frequency is converging towards its limit. To deal with this problem, CPU producers tend to develop CPUs with more and more cores which leads to a greater focus of parallelization of software development.

Meanwhile, the development of graphics processing units (GPU) is at the other end of the spectrum, with a positive performance inclination. Today, some GPUs have outpaces standard CPUs in floating point performance. This has created a trend to use this computing power to more than just rendering graphics onto the screen. The GPU can be used to perform computations with high throughput which in some cases might drastically outperform the same computations on a CPU. This is called General-purpose computing on graphics processing units (GPGPU) and multiple frameworks have been developed which perform GPGPU. The most popular ones are OpenCL, CUDA and DirectX DirectCompute.

The purpose of this thesis is to measure the capacity of OpenCL, CUDA and DirectCompute in terms of performance, scalability and how intuitive the frameworks are to work with. A algorithm will be implemented in each of the three frameworks, and will be compared to a previous implementation of FFT (Fast Fourier Transform).

This thesis will be done at MindRoad AB in Mjärdevi, Linköping, Sweden.

## Approach

- Feasibility study
  - Find and compare algorithms that are parallelizable and suitable to be implemented on GPUs
  - Study previous research on the subject
- Implement the chosen algorithm on all of the three frameworks
- Compare the implementations with each other and the previous implementation of the FFT algorithm in terms of performance, scalability and intuitivity

## Resources

- **OpenCL** - <https://www.khronos.org/opencv/>
- **CUDA** - <https://www.geforce.com/hardware/technology/cuda>
- **DirectCompute** - <https://blogs.msdn.microsoft.com/chuckw/2010/07/14/directcompute/>
- Course material from TDDD56 - Multicore and GPU Programming